The design of Virtual CID based on NTCIP

Daobin (Ben) Wang, Postdoc
Center for Advanced Transportation Education and Research
University of Nevada, Reno
Email: wangdaobin08@gmail.com
Guideline

- Background
- Design of Virtual CID
- Tests
- Contributions
- Future Works
Requirement 1: Transportation Engineers want to estimate a timing plan before input it into a signal controller.

Requirement 2: A simulation software needs to communicate with a real controller.

Requirement 3: There are just few ways to connect the simulation software to the signal controller; but the costs are high.
Current Ways: Some companies have developed the “Hardware in Loop” to achieve it. (e.g., the Test box and the CID)
Current Ways: Some companies have developed the “Hardware in Loop” to achieve it. (e.g., the Test box and the CID)

<table>
<thead>
<tr>
<th>Test box</th>
<th>CID</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Disadvantages</strong></td>
<td></td>
</tr>
<tr>
<td>One controller needs a corresponding test box or a CID.</td>
<td></td>
</tr>
<tr>
<td>Complicated for operation</td>
<td></td>
</tr>
<tr>
<td>Inconvenience</td>
<td></td>
</tr>
<tr>
<td>High costs</td>
<td></td>
</tr>
</tbody>
</table>
Goal: to develop a software that can directly connect the simulation software to the controller.
Design

Prepared Materials:

- NTCIP Protocol
- COM Protocol
- Testbox Protocol

---

1. The CID / Testbox can act as either a TLS Cabinet (3530 and MNI) or a 3530 Cabinet (3070-TDS).
2. The CID learns of what mode it is operating by being initialized at power up by the PC. Additionally, during initialization, other aspects of the CID are setup as well, such as the virtualized MNI programming card.
3. To make the CID simple and fast, the data sent to the CID is actually the FSD or BSO response that the controller will receive. In other words, you do not send detector status to the CID, instead you send the CID the message that the detector just will send to the controller when triggered. This is important, because not only does this make the CID a simple device, but it also allows you PC side application to be very simple.
4. Serial port is 25600 Bps 8N1
5. Message format:

   [0] HEADER BYTE (0x0F)
   [1] COMMAND BYTE
   [2..X] DATA BYTES
   [X+1] TWO’S-COMPLEMENT CHECKSUM

6. COMMAND [0]: Initialization

   [0] HEADER BYTE (0x0F)
   [1] COMMAND BYTE (0x5C)
   [2..X] DATA BYTES (aligned per TLS spec)
   [X] OPTION BYTES
   X 0 1 0 0 0 0
   SEND TO CID 0-MIN, 1-YES
   X A B C D DEVICE TYPE (0=TLS, 1=PTVISA, 2=PTVISE)
   [X+1] TWO’S-COMPLEMENT CHECKSUM
Design

- Framework

Controller

- Serial Ports
- Ethernet

Naztec Test Box Protocol

NTCIP Protocol

CCM

Processing Module

Simulation Speed Control

Thread 1

Thread 2

Watch Dog

COM Protocol

VCM

Setting Module

VISSIM

Main

Controller
Design

- Demo (https://www.youtube.com/watch?v=cUXPGaBPkyY)
Stability test: This program can run at least 3 hours without any problem when other devices work well. (30 times)

<table>
<thead>
<tr>
<th>Run (Ethernet)</th>
<th>Result</th>
<th>Run (Test box)</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Pass</td>
<td>1</td>
<td>Pass</td>
</tr>
<tr>
<td>2</td>
<td>Pass</td>
<td>2</td>
<td>Pass</td>
</tr>
<tr>
<td>3</td>
<td>Pass</td>
<td>3</td>
<td>Pass</td>
</tr>
<tr>
<td>4</td>
<td>Pass</td>
<td>4</td>
<td>Pass</td>
</tr>
<tr>
<td>5</td>
<td>Pass</td>
<td>5</td>
<td>Pass</td>
</tr>
<tr>
<td>6</td>
<td>Pass</td>
<td>6</td>
<td>Pass</td>
</tr>
<tr>
<td>7</td>
<td>Pass</td>
<td>7</td>
<td>Pass</td>
</tr>
<tr>
<td>8</td>
<td>Pass</td>
<td>8</td>
<td>Pass</td>
</tr>
<tr>
<td>9</td>
<td>Pass</td>
<td>9</td>
<td>Pass</td>
</tr>
<tr>
<td>10</td>
<td>Pass</td>
<td>10</td>
<td>Pass</td>
</tr>
<tr>
<td>11</td>
<td>Pass</td>
<td>11</td>
<td>Pass</td>
</tr>
<tr>
<td>12</td>
<td>Pass</td>
<td>Fail:Cable Loose</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Pass</td>
<td>13</td>
<td>Pass</td>
</tr>
<tr>
<td>14</td>
<td>Pass</td>
<td>14</td>
<td>Pass</td>
</tr>
<tr>
<td>15</td>
<td>Pass</td>
<td>15</td>
<td>Pass</td>
</tr>
</tbody>
</table>
Compatibility test: This program can only support the Econolite Cobalt Controller (using the Ethernet), and can support various kinds of controllers when using the Naztec test box.

<table>
<thead>
<tr>
<th>Ethernet</th>
<th>Test box</th>
</tr>
</thead>
<tbody>
<tr>
<td>Econolite Cobalt</td>
<td>Econolite Cobalt</td>
</tr>
<tr>
<td>Naztec Series 900</td>
<td></td>
</tr>
<tr>
<td>Eagle EPAC 300</td>
<td></td>
</tr>
</tbody>
</table>
Contribution 1: It can connect a simulation software to multiple controllers without hardware.

Contribution 2: It can support the Testbox connection.

Contribution 3: Convenient for field application

Contribution 4: User friendly interface
Contributions

- Contribution 1: It can connect a simulation software to multiple controllers without hardware.

Important

There is no previous software in the United States; we are the pioneer.
Future work

- Add the timing plan setting function.
- Improve the accuracy of simulation speed control.
Thank You